

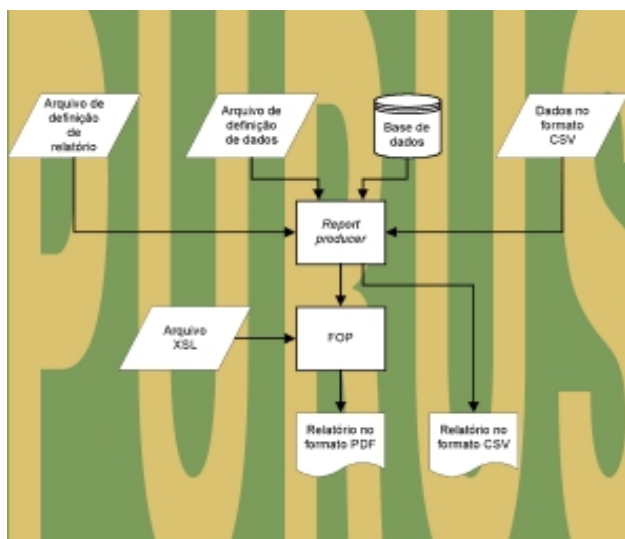
Gerador de Relatórios Purus

Marcos Cezar Visoli¹

Moacir Pedroso Júnior²

João Francisco Gonçalves Antunes³

Pedro Henrique Albertini⁴



Purus é um gerador de relatórios, desenvolvido pela Embrapa Informática Agropecuária, sob a plataforma J2EE (Sun Microsystems, 2002), que possibilita a elaboração de relatórios, a partir de informações oriundas de banco de dados ou arquivos no formato CSV (*Comma Separated Value*) (Creativyst, 2002), dispostos em tabelas ou listas. Ainda, possui recursos como agrupamento de dados, quebra de página, funções de agrupamento e formatação de campos de dados. O Purus pode gerar os relatórios nos formatos PDF (*Portable Document Format*) (Adobe Systems Incorporated, 2002) ou CSV.

O Purus teve sua origem no projeto SIGI (Pedroso Júnior et al., 2001) que trata do desenvolvimento de um sistema informatizado para o gerenciamento de projetos de pesquisa, de desenvolvimento institucional e de negociação tecnológica. Ainda, suporta módulos que agrupam informações destes projetos em planos táticos e operacionais, gerenciamento de tabelas do sistema, controle de acesso, etc. É composto por uma aplicação chamada SigiCliente, que é executada na plataforma Windows (95, NT, 2000), construído em Borland Delphi, que serve para editar os projetos gerenciados pelo sistema, e uma outra parte, chamada SigiServidor, construída sob a plataforma J2EE, que contém os outros módulos do sistema e opera via Web.

No SigiServidor, a grande demanda existente para produzir relatórios, levou à identificação de ferramentas para produção de relatórios, que fosse de fácil uso e com baixo custo. Não foi encontrada qualquer uma que atendesse às necessidades, básicas, de acordo com as características apresentadas no Purus, e que não oferecesse custo adicional ao projeto. Resolveu-se então desenvolver uma ferramenta própria no projeto.

Os itens a seguir descrevem os arquivos de especificação de dados e relatórios, os principais recursos disponíveis, e instruções de uso do Purus.

Arquitetura do Purus

A Fig. 1 ilustra a arquitetura do Purus. O *Report Producer*, tem como entrada o arquivo de definição de relatórios e os arquivos de definição de dados. O primeiro define como os dados serão dispostos no relatório, a formatação de campos, etc., enquanto que o segundo (que pode ser mais de um) define de onde serão obtidos os dados para a produção do relatório. Dependendo da origem de dados utilizada no arquivo de definição de relatórios, o *Report Producer* busca os dados de uma base de dados ou de um arquivo no formato CSV. Caso o formato de saída do relatório seja CSV, o Report Producer realiza o processamento e produz a saída neste formato. Caso o formato de saída

1 Bsc. em Ciência da Computação, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo – 13083-970 – Campinas, SP. (e-mail: visoli@cnptia.embrapa.br)

2 Ph.D. em Pesquisa Operacional, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo – 13083-970 – Campinas, SP. (e-mail: pedroso@cnptia.embrapa.br)

3 Bsc. em Matemática Aplicada e Estatística, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo – 13083-970 – Campinas, SP. (e-mail: joaof@cnptia.embrapa.br)

4 Consultor da Embrapa Informática Agropecuária na área de desenvolvimento do SIGI – (e-mail: pedroalbertini@directnet.com.br)

especificado seja PDF, é produzido um arquivo XML(*eXtensible Markup Language*) (Harold & Means, 2000) intermediário para o FOP (*Formatting Object Processor*) (Apache Software Foundation, 2002).

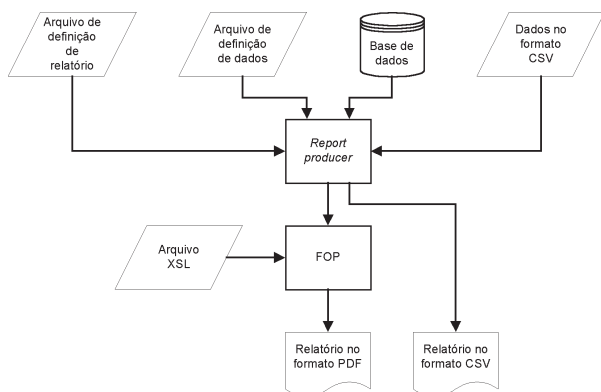


Fig. 1. Arquitetura do Purus.

O FOP é um transformador XSL (*eXtended Style Language*) e, através de um arquivo no formato XSL, processa o arquivo enviado pelo Report Producer e finalmente produz o relatório no formato PDF.

Segundo Pedroso Júnior et al. (2002), tem-se uma descrição mais detalhada do Purus, incluindo a especificação dos arquivos intermediários.

Arquivo de Definição de Dados (*DataSource*)

O arquivo de definição de dados, no formato padrão XML, contém toda a informação necessária para o Purus buscar os dados de um sistema gerenciador de bases de dados ou de um arquivo CSV. Cada arquivo pode conter uma ou mais definições de *DataSource*, e também, mais de um arquivo pode ser referenciado. Assim, o Purus oferece uma grande versatilidade para a organização dos *DataSources*.

A seguir é apresentada a relação dos elementos que fazem parte de um arquivo de definição de dados, juntamente com a sua descrição:

Elemento	Definição
Data	Elemento raiz na definição de um <i>datasource</i> . Contém um ou mais <i>datasources</i> . Não possui atributos.
DataSource	Contém as informações necessárias para o Purus buscar os dados de uma conexão com um banco de dados ou arquivo CSV. Atributos: <i>Name</i> : usado para uma posterior referência ao <i>datasource</i> . <i>type</i> : indica o tipo de fonte de dados. Pode assumir os valores csv ou sql.
Location	Contém o caminho completo do arquivo CSV que contém os dados. Necessário somente se o atributo <i>type</i> for csv. Atributo: <i>url</i> : caminho completo do arquivo CSV que contém os dados.
DbConnection	Necessário apenas se o atributo <i>type</i> for sql. Contém a informação necessária para abrir uma conexão JDBC (<i>Java Database Connectivity</i>) com o banco de dados. Atributos: <i>Username</i> : nome do usuário para conectar no banco de dados. <i>Password</i> : senha. <i>Driver</i> : driver JDBC a ser usado. <i>url</i> : endereço do banco.
DbQuery	Contém o comando de seleção, com a sintaxe SQL (W3Schools.com, 2002), usado para buscar informações de um banco de dados. Atributos: <i>DbConnection</i> : nome do DbConnection a ser usado.
Field	Define uma coluna de dados. Atributos: <i>Name</i> : nome do campo, para ser referenciado na definição de relatório. <i>Index</i> : associa o campo ao seu equivalente em uma lista de resultados do comando de seleção dos dados. <i>type</i> : classe java equivalente ao tipo de dado desse campo.

Arquivo de definição de relatório

O arquivo de definição de relatório contém toda a informação necessária para que o Purus realize a exibição dos dados obtidos a partir do arquivo de definição de

dados. Cada arquivo de definição de relatório pode ter apenas uma definição de relatório, embora possa referenciar vários arquivos de definição de dados.

A seguir são apresentados os elementos disponíveis:

Elemento	Definição
Report	Elemento raiz na definição de um relatório. Não possui atributos.
FrontPage	Definição da primeira página do relatório. Pode conter uma ou mais linhas. Não possui atributos.
Line	Define uma linha de texto. Atributos: <i>Spacebefore</i> : espaçamento vertical antes da linha. <i>Default</i> ="6pt". <i>Alignment</i> : alinhamento do texto. <i>Default</i> ="left". <i>Fontfamily</i> : (Arial, Times New Roman, etc). <i>Default</i> ="Arial". <i>Fontweight</i> : (bold, bolder, light, lighter, normal). <i>Default</i> ="normal". <i>Fontsize</i> : tamanho do fonte em cm, mm ou pt. <i>Default</i> ="10pt". <i>Fontstyle</i> : estilo do fonte, como normal ou italic. <i>Default</i> ="normal". <i>Fontcolor</i> : cor do texto. <i>Default</i> ="black". <i>Backgroundcolor</i> : cor do fundo. <i>Default</i> ="white". Quando o atributo não é definido, seu valor <i>Default</i> será usado.
Header	Define o cabeçalho, que aparecerá na parte superior de todas as páginas do relatório. Pode conter um ou mais elementos <i>Line</i> , por exemplo, <header> <line> ... </line> <line> ... </line> </header> Não possui atributos.
Footer	Define o rodapé, que aparecerá na parte inferior de todas as páginas do relatório. Atributos: <i>Creationtime</i> : indica se a data de criação do relatório deve ou não ser mostrada. Os possíveis valores são <i>true</i> ou <i>false</i> .
Section	Define uma seção do relatório. O relatório pode ter várias seções, cada uma associada com <i>datasources</i> distintos. Atributos: <i>Datasource</i> : nome do <i>datasource</i> de onde será obtido os dados dessa seção. <i>Newpage</i> : indica se haverá quebra de página antes da seção. Os possíveis valores são <i>true</i> ou <i>false</i> . <i>type</i> : tipo da seção, indica a forma de apresentação dos dados. Os valores para o atributo são <i>column</i> ou <i>row</i> . <i>Style</i> : estilo da seção, usado quando o atributo <i>type</i> é <i>row</i> . Pode ser <i>listing</i> ou <i>table</i> .
Title	Define o título de uma seção, que será exibido apenas uma vez, antes dos dados. O conteúdo do título é especificado através de comandos <i>Line</i> , um para cada linha. O elemento <i>Title</i> deve conter pelo menos um elemento <i>Line</i> . Não possui atributos.
Column	Define uma coluna de dados na seção. Atributos: <i>Spacebefore</i> : define o deslocamento horizontal antes da coluna em cm, mm ou pt <i>Maxlabelwidth</i> : define o tamanho máximo de um rótulo em cm, mm ou pt. Usado para seções com o atributo <i>type row</i> e atributo <i>style table</i> . <i>Maxdatawidth</i> : define o tamanho máximo de uma coluna de dados, em cm, mm ou pt. Usado para seções com o atributo <i>type row</i> e atributo <i>style table</i> .
Label	Define o formato do rótulo da coluna. Atributos: <i>Value</i> : descrição do rótulo Os atributos <i>alignment</i> , <i>fontfamily</i> , <i>fontweight</i> , <i>fontsize</i> , <i>fontstyle</i> , <i>fontcolor</i> e <i>backgroundcolor</i> são idênticos ao descrito no elemento <i>Line</i> .
Data	Define a fonte de dados da coluna e o formato do texto. Atributos: <i>Source</i> : associa a coluna com o campo de dados (field) de um <i>datasource</i> descrito no arquivo de definição de dados. <i>Width</i> : define o tamanho horizontal da coluna. Os atributos <i>alignment</i> , <i>fontfamily</i> , <i>fontweight</i> , <i>fontsize</i> , <i>fontstyle</i> , <i>fontcolor</i> e <i>backgroundcolor</i> são idênticos ao descrito no elemento <i>Line</i> .

Exemplo

Segue o exemplo de uma saída no formato PDF e o conteúdo do arquivo de definição de um relatório que o produziu.



The screenshot shows a PDF report generated by Purus. It features a green logo in the top left, a main title 'Relatório de Funcionários' in the top right, and a subtitle 'Funcionários' in the center. Below the subtitle is a table with 5 columns: Nome, Departamento, E-mail, Idade, and Salario. The table contains 10 rows of employee data. At the bottom left, there is a 'creation time' stamp indicating the report was created on 15-Jul-02. At the bottom right, there is a footer with the title 'Relatório de Funcionários' and the page number '2'.

Nome	Departamento	E-mail	Idade	Salario
Bianca	Diretoria	bianca@lalala.com.de	19	RS 50.000,00
Debora	Administração	debora@debby.net	23	RS 45.000,00
Marina	Cozinha	mary@zipmail.com.br	27	RS 30.000,00
Carol	Vice-Presidencia	carol@email.la.la	17	RS 90.000,00
Arthur	RH	arthur@ibest.com.br	35	RS 15.000,00
Carlos	Vendas	carlos@hotmail.com	35	RS 10.000,00
Danilo	Informatica	daniel@ig.com.br	20	RS 5.000,00
PedroH	Presidencia	pedroh@cnptia.embrapa.br	19	RS 99.999,00
Rafael	Limpeza	rafael@blizzard.com	30	RS 10.000,00

```
<Report>
  <FrontPage>
    <Line fontsize="80pt" spacebefore="100pt" alignment="center">Relatório</Line>
    <Line fontsize="80pt" alignment="center">de</Line>
    <Line fontsize="80pt" alignment="center">Funcionários</Line>
  </FrontPage>

  <Header>
    <Line fontweight="bold" alignment="center" fontsize="25pt">
      Relatório de Funcionários
    </Line>
  </Header>

  <Footer>
    <Line>Relatório de Funcionários</Line>
  </Footer>

  <Section datasource="ds_funcionarios" type="column" style="">
    <Title>
      <Line fontsize="30pt" alignment="center">
        Funcionários
      </Line>
    </Title>
    <Column spacebefore="1cm" maxlabelwidth="4cm" maxdatawidth="11.5cm">
      <Label fontsize="11pt" fontcolor="white" backgroundcolor="green" value="Nome"
width="4cm"/>
      <Data fontsize="9pt" source="nome"/>
    </Column>
    <Column>
      <Label fontsize="11pt" fontcolor="white" backgroundcolor="green" value="Departamento"
width="3cm"/>
      <Data fontsize="9pt" source="departamento"/>
    </Column>
    <Column>
      <Label fontsize="11pt" fontcolor="white" backgroundcolor="green" value="E-mail"
width="4cm"/>
      <Data fontsize="9pt" source="email"/>
    </Column>
    <Column>
      <Label fontsize="11pt" fontcolor="white" backgroundcolor="green" value="Idade"
width="1.5cm"/>
      <Data fontsize="9pt" source="idade"/>
    </Column>
    <Column>
      <Label fontsize="11pt" fontcolor="white" backgroundcolor="green" value="Salario"
width="2.0cm"/>
      <Data fontsize="9pt" alignment="right"
source="salario.formatCurrency('br','portuguese')"/>
    </Column>
  </Section>
</Report>
```

Recursos

Tipos de seção

Os dados de uma seção podem ser dispostos em dois formatos principais, coluna ou linha, de acordo com o atributo *type*. No formato coluna, as informações são exibidas como uma tabela, com um cabeçalho, onde estão os rótulos das colunas, e a seguir os dados alinhados verticalmente aos rótulos.

type="column" style=""

Nome	Telefone
TIAGO	32345678
PEDRO	32939819
JOSÉ	33564678

type="row" style="listing"

Nome TIAGO
Telefone 32345678
Nome PEDRO
Telefone 32939819
Nome JOSE
Telefone 33564678

type="row" style="table"

Nome:	TIAGO
Telefone:	32345678
Nome:	PEDRO
Telefone:	32939819
Nome:	JOSE
Telefone:	33564678

No formato linha os dados são exibidos como uma lista, com apenas uma coluna de dados por linha. Para este formato, existem duas variações, lista (*listing*) ou tabela (*table*) definidos pelo atributo *style*.

A seguir são apresentados exemplos dos tipos de seção disponíveis, com os respectivos estilos de acordo com os atributos disponíveis.

Agrupamento

O agrupamento de dados é um recurso importante do Purus que permite produzir relatórios organizados adequadamente quando, por exemplo, as informações possuem dependência hierárquica. Para isto, é usado como critério uma coluna também especificada no arquivo de definição de relatório.

A definição dos itens de agrupamento é realizada pelo elemento *Group*. Pode haver mais de uma


coluna que agrupa os dados, mas elas devem sempre obedecer uma hierarquia, que é importante para o uso de funções de grupo. O elemento *Group* deve conter um elemento *Label* e um elemento *Data*, que definem por qual campo é realizado o agrupamento e, pelas colunas dos campos restantes.

A seguir temos o exemplo de definição de grupo, com o trecho de um arquivo de definição de relatório, e posteriormente é apresentado o resultado.

```
<Section datasource="ds_funcionarios" type="column" style="">
  <Group>
    <Label fontsize="16pt" fontcolor="black" fontweight="bold" backgroundcolor="white"
value="SEXO" width="1.8cm">
      <Data fontsize="14pt" source="sexo"/>
      <Column spacebefore="1cm">
        <Label fontsize="11pt" fontcolor="white" backgroundcolor="green" value="Nome"
width="4cm"/>
        <Data fontsize="9pt" source="nome"/>
      </Column>
      <Column>
        <Label fontsize="11pt" fontcolor="white" backgroundcolor="green"
value="Departamento" width="3cm"/>
        <Data fontsize="9pt" source="departamento"/>
      </Column>
      ...
    </Group>
```

No exemplo, o elemento na linha destacada identifica o campo pelo qual é realizado o agrupamento. A seguir

é apresentado um trecho do relatório resultante da especificação mostrada.



Relatório de Funcionários

group
data

↓

Funcionários

group
label →

SEXO Feminino

Nome	Departamento	E-mail	Idade	Salario
Bianca	Diretoria	bianca@lalala.com.de	19	R\$ 50.000,00
Debora	Administração	debora@debby.net	23	R\$ 45.000,00
Marina	Cozinha	mary@zipmail.com.br	27	R\$ 30.000,00
Carol	Vice-Presidencia	carol@email.la.la	17	R\$ 90.000,00

SEXO Masculino

Nome	Departamento	E-mail	Idade	Salario
Arthur	RH	arthur@ibest.com.br	35	R\$ 15.000,00
Carlos	Vendas	carlos@hotmail.com	35	R\$ 10.000,00
Danilo	Informatica	daniel@ig.com.br	20	R\$ 5.000,00
PedroH	Presidencia	pedroh@cnptia.embrapa.br	19	R\$ 99.999,00
Rafael	Limpeza	rafael@blizzard.com	30	R\$ 10.000,00

Creado em 15-jul-02
Relatório de Funcionários
2

Funções de grupo

Com o recurso de possibilitar o agrupamento de informações, o Purus disponibiliza as seguintes funções que podem ser usadas nesses blocos de dados:

- *count* – número de itens
- *avg* – média aritmética dos valores de uma coluna
- *sum* – total dos valores de uma coluna
- *min* – menor valor dos valores de uma coluna
- *max* – maior valor dos valores de uma coluna

As funções *avg*, *sum*, *min* e *max* somente podem ser usadas para campos (*datafields*) do tipo `java.lang.Float`.

O item *Functions* é utilizado para agrupar as funções que necessitam ser utilizadas no relatório. No seu escopo estão definidas as funções, cada uma pelo item *Function*.

Deve ser observado que o item *Functions* deve ser colocado logo depois do item *Group* ou *Section*. Se um dos grupos do nível inferior da hierarquia contiver o item *Functions*, os níveis acima devem, necessariamente, também especificar o item *Functions*.

A seguir tem-se um relatório com agrupamento, e logo após o trecho de um arquivo com o exemplo da definição de relatórios com a especificação de funções de grupos.



Relatório de Funcionários

Funcionários

SEXO Feminino

Nome	Departamento	E-mail	Idade	Salario
Bianca	Diretoria	bianca@lalala.com.de	19	R\$ 50.000,00
Debora	Administração	debora@debby.net	23	R\$ 45.000,00
Marina	Cozinha	mary@zipmail.com.br	27	R\$ 30.000,00
Carol	Vice-Presidencia	carol@email.la.la	17	R\$ 90.000,00
Qtde:	Idade Media	Total Salário:		
4.0	21.5	215000.0		

SEXO Masculino

Nome	Departamento	E-mail	Idade	Salario
Arthur	RH	arthur@ibest.com.br	35	R\$ 15.000,00
Carlos	Vendas	carlos@hotmail.com	35	R\$ 10.000,00
Danilo	Informatica	daniel@ig.com.br	20	R\$ 5.000,00
PedroH	Presidencia	pedroh@cnptia.embrapa.br	19	R\$ 99.999,00
Rafael	Limpeza	rafael@blizzard.com	30	R\$ 10.000,00
Qtde:	Idade Media	Total Salário:		
5.0	27.8	139999.0		

Qtde:	Idade Media	Total Salário:
9.0	25.0	354999.0

```

...
<Group | Section>
...
<Functions type="column">
  <Function type="count" column="nome" spacebefore="1cm">
    <Data fontcolor="black" backgroundcolor="black"/>
    <Label fontcolor="white" fontweight="bold" backgroundcolor="green" value="Qtde:"
width="3.0cm"/>
  </Function>
  <Function type="avg" column="idade" maxlabelwidth="4cm">
    <Data alignment="center"/>
    <Label fontcolor="white" fontweight="bold" backgroundcolor="green" value="Idade
Media" width="2.5cm"/>
  </Function>
  <Function type="sum" column="salario" maxlabelwidth="4cm">
    <Data alignment="center"/>
    <Label fontcolor="white" fontweight="bold" backgroundcolor="green" value="Total
Salário:" width="2.5cm"/>
  </Function>
</Functions>
...

```

Parametrização de definição de relatório

Em alguns relatórios faz-se necessário parametrizar alguns valores, para que, dado um conjunto diferente de dados, não seja necessário recriar um novo arquivo de definição. Por exemplo, em um caso onde é necessário produzir um relatório de funcionários para cada departamento, é possível especificar um parâmetro para o título do relatório, substituindo-o em cada chamada do Purus (mais adiante é descrito como realizar a chamada do Purus) pelo nome do departamento. Com o uso de parâmetros não é necessário elaborar um relatório para cada departamento.

A sintaxe para o uso de parâmetros é:

`$(nome |default)`

onde:

`nome`: nome do parâmetro.

`default`: valor a ser exibido, caso este não seja definido na chamada do Purus.

Inclusão de caracteres ASCII

O interpretador utilizado pelo Purus, em alguns lugares, interpreta o espaço como quebra de linha. Para contornar essa situação, nos lugares onde é necessário, deve-se utilizar no lugar do espaço o código “ ”. 160 é o código ASCII do caracter “nbsp” (*non breakable space*). O não pode ser utilizado pois um dos pacotes que compõem o Purus não o interpreta como espaço em branco. Essa técnica pode ser utilizada para inserir qualquer caracter da tabela ASCII.

Formatação de dados

Existem algumas funções disponíveis no Purus para formatação dos dados do relatório. Essas funções devem ser especificadas na definição dos elementos *data* das colunas desejadas.

As funções existentes no Purus são:

- `trim()` – retira espaços em branco anteriores e posteriores do valor do campo;
- `toUpper()` – transforma os caracteres do valor do campo para maiúsculo;
- `insertParenteses()` – deixa o valor do campo entre parênteses;
- `insertBefore(String text)` – insere o texto *text* antes do início do valor do campo;

- `insertAfter(String text)` – insere o texto *text* após o valor do campo;
- `formatNumber(String language,String country)` – formata o campo, neste caso número, para o formato padrão do país *country* e idioma *language*;
- `formatCurrency(String language,String country)` – formata o campo, neste caso número monetário, para o padrão do país *country* e idioma *language*;
- `formatDate(String outputPattern)` – formata o campo, neste caso data, para o formato defini-

```
<Column>
  <Label fontsize="11pt" fontcolor="white"
backgroundcolor="green" value="Salário"
width="2.0cm"/>
  <Data fontsize="9pt" alignment="right"
source="salario.formatCurrency('br','portuguese')">
</Column>
```

do por *outputPattern*.

O trecho seguinte apresenta um exemplo de uso de funções:

As funções usadas para formatar são obtidas da classe *Formatter*, usando o método de reflexão disponível na linguagem java. Novas funções podem ser adicionadas, o que possibilita que o Purus possa ser evoluído.

Relatórios no formato CSV

Como citado anteriormente, o Purus também produz relatórios no formato CSV, possibilitando mais uma diversidade de relatórios, pois a partir de arquivos deste formato, pode-se utilizar outras ferramentas para produzir outros relatórios customizados.

O padrão do Purus é produzir relatórios no formato PDF. Para produzir relatórios (ou arquivos) no formato CSV, é necessário informar esta opção como parâmetro na invocação do Purus.

Execução do Purus

ReportManager é a principal classe do Purus, que deve ser usada para executar o Purus e produzir os relatórios. A sintaxe do comando para executar o Purus é:

java – classpath {classpath} ReportManager [parâmetros]

onde:

- classpath:
caminho das bibliotecas e das classes do Purus
- parâmetros:
 - input:<arquivo de entrada> Arquivo de definição de relatório (XML)
 - output:<arquivo de saída> Arquivo CSV ou PDF que será produzido
 - style:<arquivo XSL> Arquivo XSL que será usado pelo *Transformer* para gerar o FO
 - data:<arquivo de dados> Arquivo(s) de definição de dados (XML)
 - d<varname>=<value> Definição de variáveis
 - type: <tipo> Tipo de arquivo a ser gerado (CSV ou PDF)

Conclusões

O Purus é utilizado no projeto SIGI desde outubro de 2001, e desde lá, vem ganhando novas funcionalidades e maior robustez, garantindo assim, uma boa ferramenta para a produção de relatórios com a complexidade exigida pelo projeto. O desenvolvimento de uma ferramenta própria para a produção de relatórios, apesar de exigir um período de tempo para sua concepção e construção, propiciou o domínio e a flexibilidade de inserir os recursos necessários, adaptando-os e evoluindo-os de acordo com novos requisitos exigidos pelo sistema ou pelo usuário.

Apesar dos recursos disponíveis serem suficientes para a produção de relatórios simples que atendem aos requisitos do SIGI, novos recursos podem ser adicionados. Porém, o objetivo na concepção do Purus, não era torná-lo o mais genérico possível, mas deixá-lo tão simples e eficiente quanto possível para atender as necessidades do projeto. A incorporação de novas funcionalidades poderiam atender novos requisitos caso o Purus venha a ser utilizado por outros projetos.

Referências Bibliográficas

ADOBE SYSTEMS INCORPORATED. **Adobe PDF**. Disponível em: <<http://www.adobe.com/products/acrobat/adobepdf.html>>. Acesso em: 4 dez. 2002.

APACHE SOFTWARE FOUNDATION. **The Apache XML Project**: FOP. Disponível em: <<http://xml.apache.org/fop>>. Acesso em: 5 dez. 2002.

CREATIVYST. **CSV - Comma Separated Value file format (how to)**. Disponível em: <<http://www.creativyst.com/Doc/Articles/CSV/CSV01.htm>>. Acesso em: 4 dez. 2002.

HAROLD, E. R.; MEANS, W. S. **XML in a nutshell: a desktop quick reference**. Beijing: O'Reilly & Associates, 2000. 480 p.

PEDROSO JÚNIOR, M.; ANTUNES, J. F. G.; VISOLI, M. C.; CASTRO, A. M. G.; LIMA, S. M. V. **Sistema de Informação Gerencial de Projetos de Pesquisa Agropecuária para o Instituto Nacional de Investigaciones Agrícolas da Venezuela**. Campinas: Embrapa Informática Agropecuária, 2001. 14 p. (Embrapa. Programa 12 – Automação Agropecuária. Projeto 12.2001.950). Projeto em andamento.

PEDROSO JÚNIOR, M.; VISOLI, M. C.; ANTUNES, J. F. G.; ALBERTINI, P. H. **Gerador de relatórios Purus: arquitetura**. Campinas: Embrapa Informática Agropecuária, 2002. 10 p. (Relatório interno do Projeto 12.2001.950).

SUN MICROSYSTEMS. **Java 2 platform, enterprise edition J2EE**. Disponível em: <<http://java.sun.com/j2ee>>. Acesso em: 05 dez. 2002

W3SCHOOLS.COM. **SQL tutorial**. Disponível em: <<http://www.w3schools.com/sql/>>. Acesso em: 4 dez. 2002.

Comunicado Técnico, 30

**Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)**
Av. André Tosello, 209
Cidade Universitária - "Zeferino Vaz"
Barão Geraldo - Caixa Postal 6041
13083-970 - Campinas, SP
Telefone (19) 3789-5743 - Fax (19) 3289-9594
e-mail: sac@cnptia.embrapa.br

1ª edição
2002 - on-line
Todos os direitos reservados

Comitê de Publicações

Presidente: José Ruy Porto de Carvalho
Membros efetivos: Amarindo Fausto Soares, Ivanilde Dispatto, Luciana Alvim Santos Romani, Marcia Izabel Fugisawa Souza, Suzilei Almeida Carneiro
Suplentes: Adriana Delfino dos Santos, Fábio Cesar da Silva, João Francisco Gonçalves Antunes, Maria Angélica de Andrade Leite, Moacir Pedroso Júnior

Expediente

Supervisor editorial: Ivanilde Dispatto
Normalização bibliográfica: Marcia Izabel Fugisawa Souza
Capa: Intermídia Publicações Científicas
Editoração Eletrônica: Intermídia Publicações Científicas